**Custom Training**

**Database Management Systems & Data Warehousing concepts.**

**Data Models:** Frameworks used to represent data, relationships, constraints and rules within a DBMS. There are 4 types of data models.

* Relational Data Model
* Entity Relationship Data Model
* Object Based Data Model
* Semi Structured Data model

**Uses of ER Diagram:**

* Helps in conceptualize the database.
* Helps in better understanding of the information that will be stored in the database.
* Reduces the complexity.
* Helps to describe the elements using entity relationship model.
* Allows users to get a preview of the logical structure of the database.

**Components of ER Diagram:**

* Rectangles: This entity relationship Diagram symbol represents entity type.
* Ellipses: The symbol represents attributes
* Diamonds: The symbol represents relationship types
* Lines: It links the attributes to the entity types and entity types with the other relationship types
* Primary Key: Here, it underlines the **attributes.**
* Double Ellipses: Represents multivalued attributes.

There are four types of attributes:

* Key Attribute:
* Composite Attribute: An attribute that is composed of several other attributes.
* Multivalued Attribute: Some attributes can have more than one attributes, they are known as multivalued attributes. These are represented by double ellipse.
* Derived Attribute:

**Entity:** An Entity can be a living or a nonliving component:

**Weak Entity:** An entity that relies on other entity is called weak entity. Eg.

**Types of Relationships:**

* One-to-One Relationships: When a single element of an entity is associated with a single element of another entity it is called one-to-one relationship.
* One-to-Many Relationships:
* Many-to-One relationships:
* Many-to-Many relationships:

**ER Modelling:**

* ER Modelling stands as a vital tool for designing and conceptual

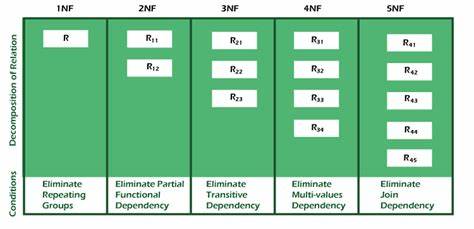
**Need for ER Modelling:**

* Data Complexity:
* Data Redundancy: duplication can lead to inconsistencies
* Data Integrity: Ensuring accuracy and validity is crucial.
* Effective Communication: ER Diagrams provide a visual representation of the DB Structure.
* Query Optimization: Understanding the relationships between the entities enhances query performance.

**Uses of ER Modelling:**

**Normalization**

* It is the process of organizing the data in the DB.
* Used to minimize the redundancy from a relation or a set of relations.
* It eliminates the undesirable characteristics like insertion, deletion, update anomalies.
* It divides the larger table into several smaller tables and links then using relationships.
* The normal forms are used to reduce redundancy from the database table.



The most basic form of data normalization is 1NF, which ensures there are no two same entries in the group. The two conditions which needs to be satisfied for 1NF are:

* Each cell should contain a single value.
* Each record should be unique.

In a 2NF table all the subsets of data that can be placed in multiple rows are placed in separate tables. It should satisfy the following rules:

* It should be in 1NF.
* The primary key should not be functionally dependent on any candidate key.

For a table to be in 3NF, it should satisfy the following conditions:

* It should be in 2NF.
* There should not be any transitive dependency.

**Data Warehouse**: Used for analytical purposes, historical data is stored. When there are multiple sources of data in such scenarios data warehouse systems come into picture.

**Fact Table**

**Dimension Table**

**Star Schema**

**Snowflake Schema**

**Fact Constellation Schema**

* **Rapidly Changing Dimensions (RCD):** these have attributes where values will be changing often.
* **Slowly Changing Dimensions:** Deals with managing changes to dimension data over time. Can be categorized as type1, type2, type3 and more based on how historical changes to dimension data are handled.
  + **Type -1: overwrite:** old data is simply overwritten with the new data when the changes occur. Historical data is lost, and the dimension table reflects the latest state of the data. Source is also changed
  + **Type-2: :**
* **Unchanging Dimensions:**

**Employee(Name, DOB, Gender, Locations, contactNumber)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EName** | **Edob** | **Egender** | **Location** | **Phone** |
|  |  |  |  |  |
| Ram | 01-01-1990 | Male | Delhi | 798573829 |
| Sunita | 02-02-1991 | Female | Bangalore | 473639100 |
| Ajay | 01-02-1985 | Male | Hyderabad | 438787821 |
| Naveen | 05-02-1991 | Male | Chennai | 444567654 |

D\_Employee/Demployee/ Dim\_employee/ DimEmployee

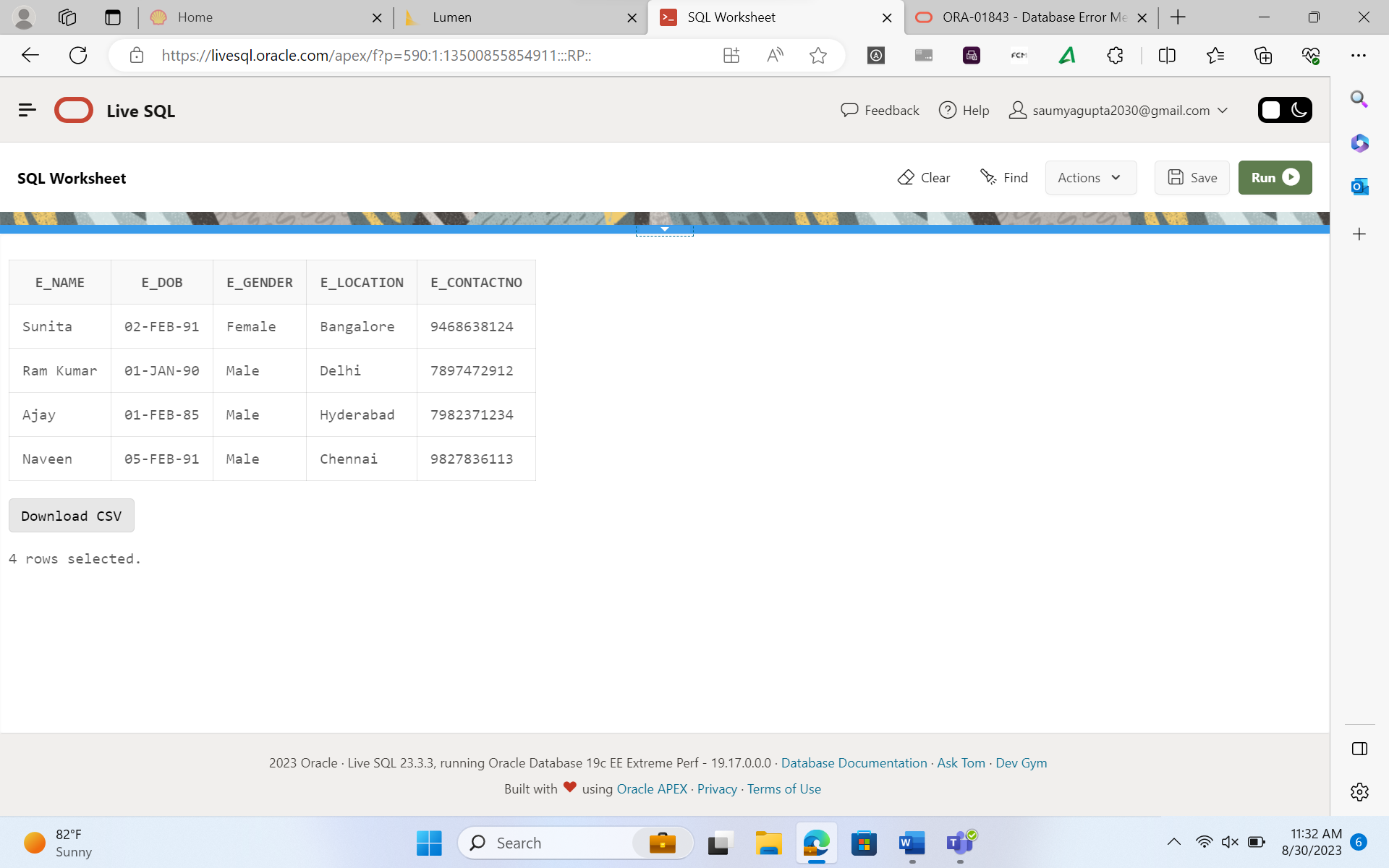
PK\_Employee\_key, E\_Contact number, name, DOB, Gender, Created Date, Updated Date -> This is how it’ll be saved in the data warehouse.

Last two cols – audit columns

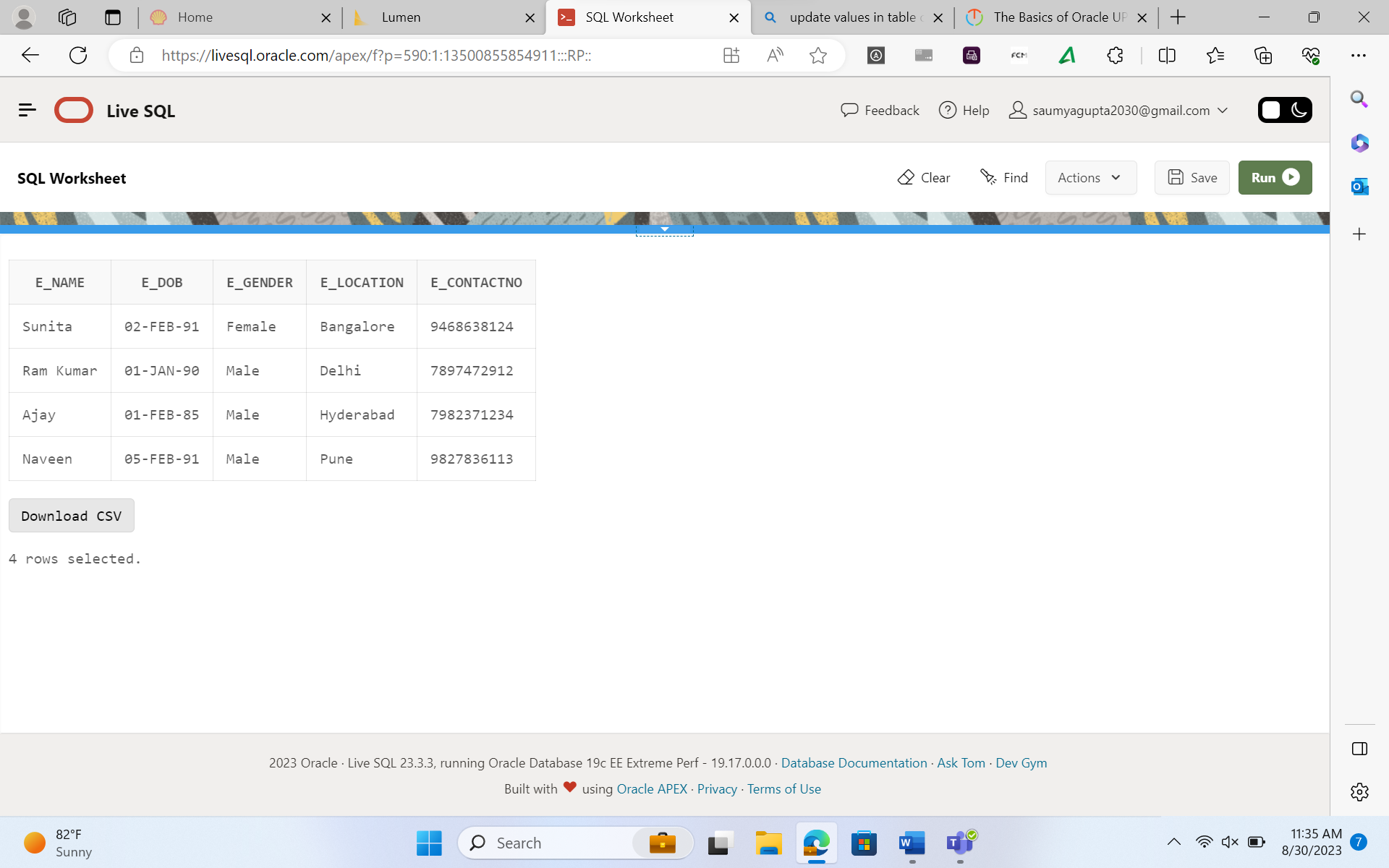
Second col – business key

What is natural key??

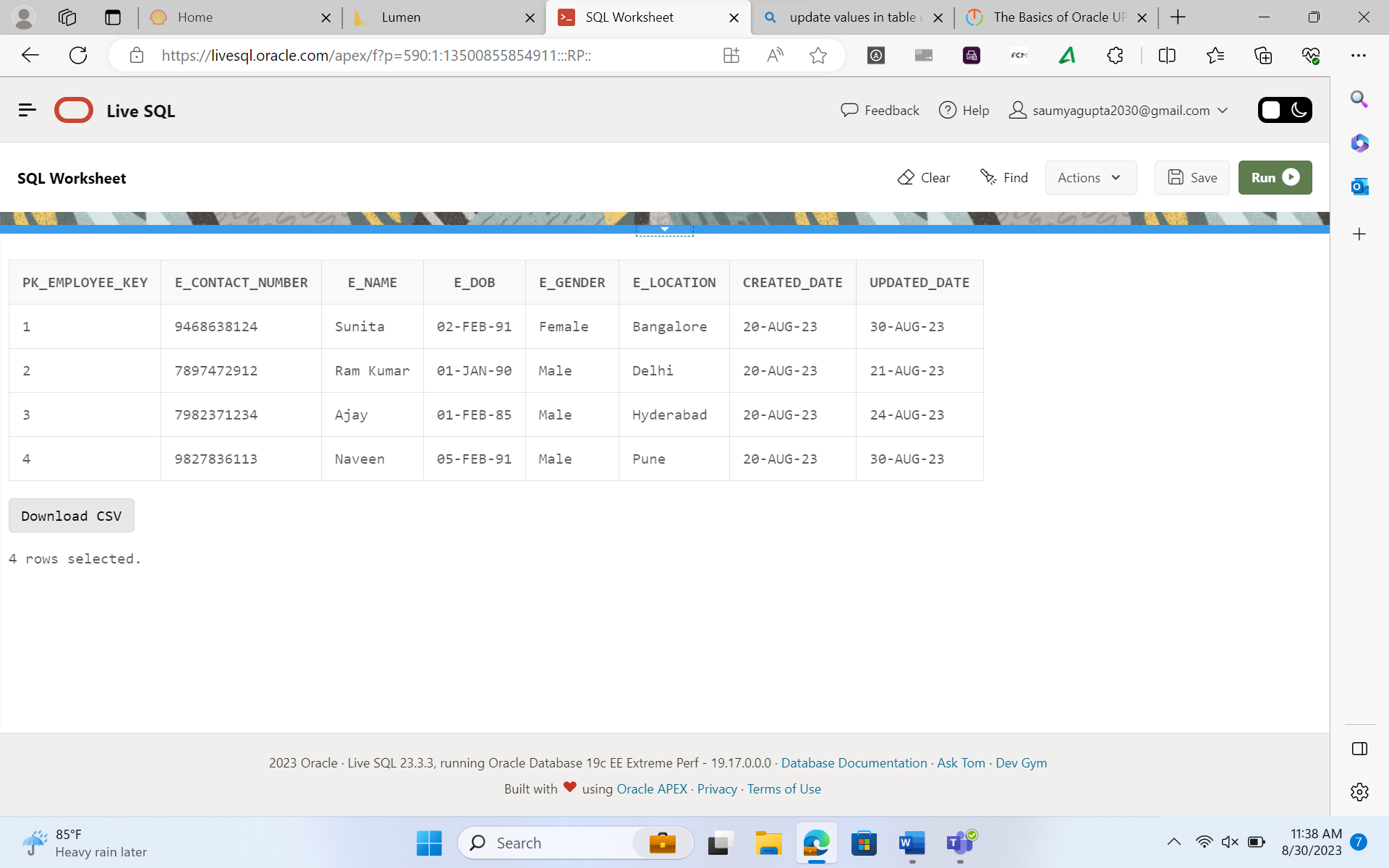
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **PK** | **Phone** | **Name** | **DOB** | **Gender** | **Location** | **Date Created** | **Date Updated** |
| 1 | 798573829 | Ram | 01-01-1990 | Male | Delhi | 30-08-2023 10:03 | 31-08-2023 10:03 |
| 2 | 473639100 | Sunita | 02-02-1991 | Female | Bangalore | 30-08-2023 10:03 |  |
| 3 | 438787821 | Ajay | 01-02-1985 | Male | Hyderabad | 30-08-2023 10:03 |  |
| 4 | 444567654 | Naveen | 05-02-1991 | Male | Chennai | 30-08-2023 10:03 |  |

****

update emp set E\_location = 'Pune' where E\_contact = 9827836113;



update D\_emp set E\_location = 'Pune', Updated\_date = '30-aug-2023' where E\_contact\_number = 9827836113;



Create table D\_emp\_SCD2(PK\_Employee\_Key number primary key,

E\_Contact\_Number number not null,

e\_name varchar(50),

e\_dob date,

e\_gender varchar(15),

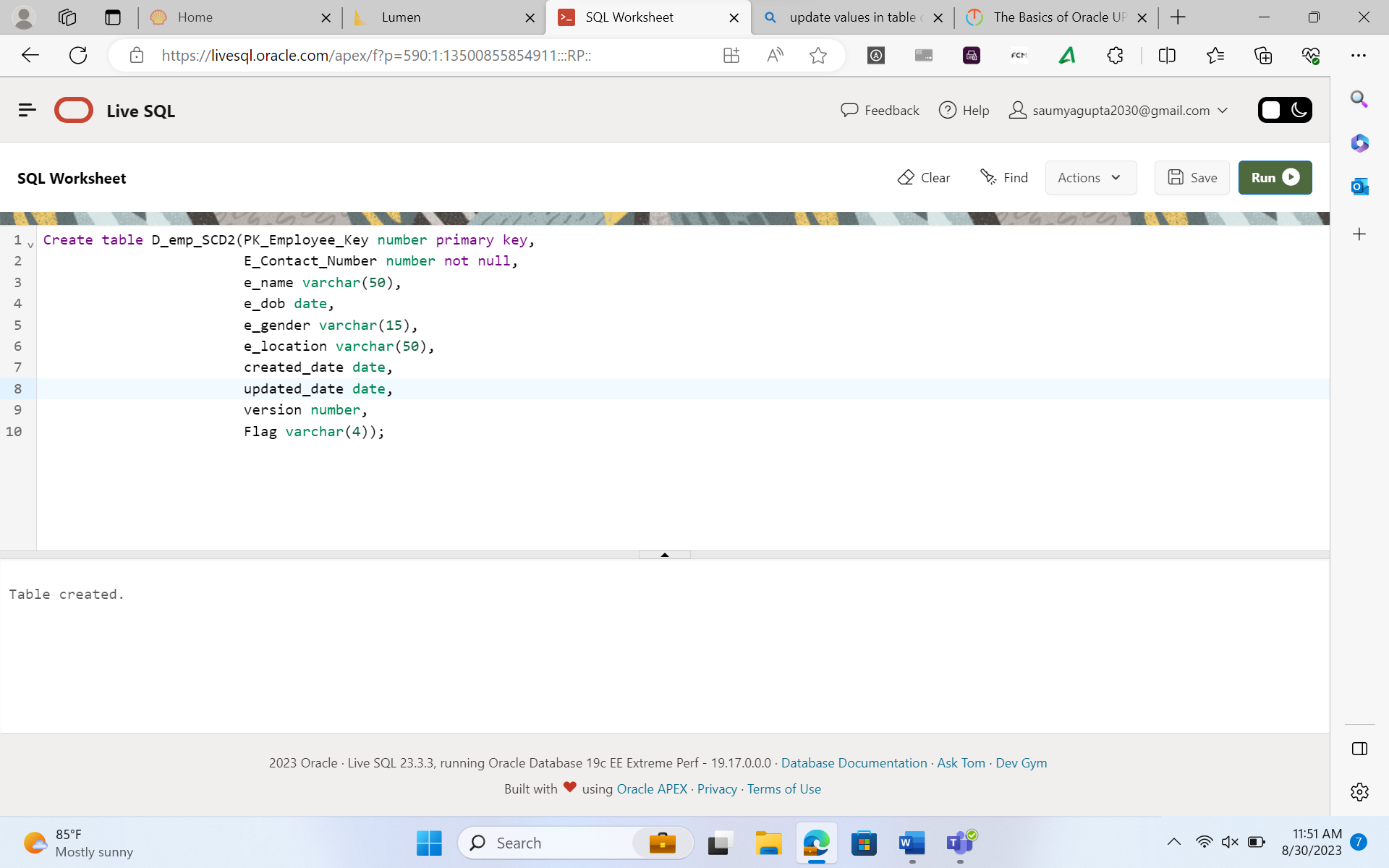
e\_location varchar(50),

created\_date date,

updated\_date date,

version number,

Flag varchar(4));



Create table D\_emp\_SCD2(PK\_Employee\_Key number primary key,

E\_Contact\_Number number not null,

e\_name varchar(50),

e\_dob date,

e\_gender varchar(15),

e\_location varchar(50),

created\_date date,

updated\_date date,

version number,

Flag varchar(4));

insert into D\_emp\_SCD2 values(1, 9468638124, 'Sunita', '02-feb-1991','Female', 'Bangalore','20-aug-2023', '21-aug-2023', 1, 'N');

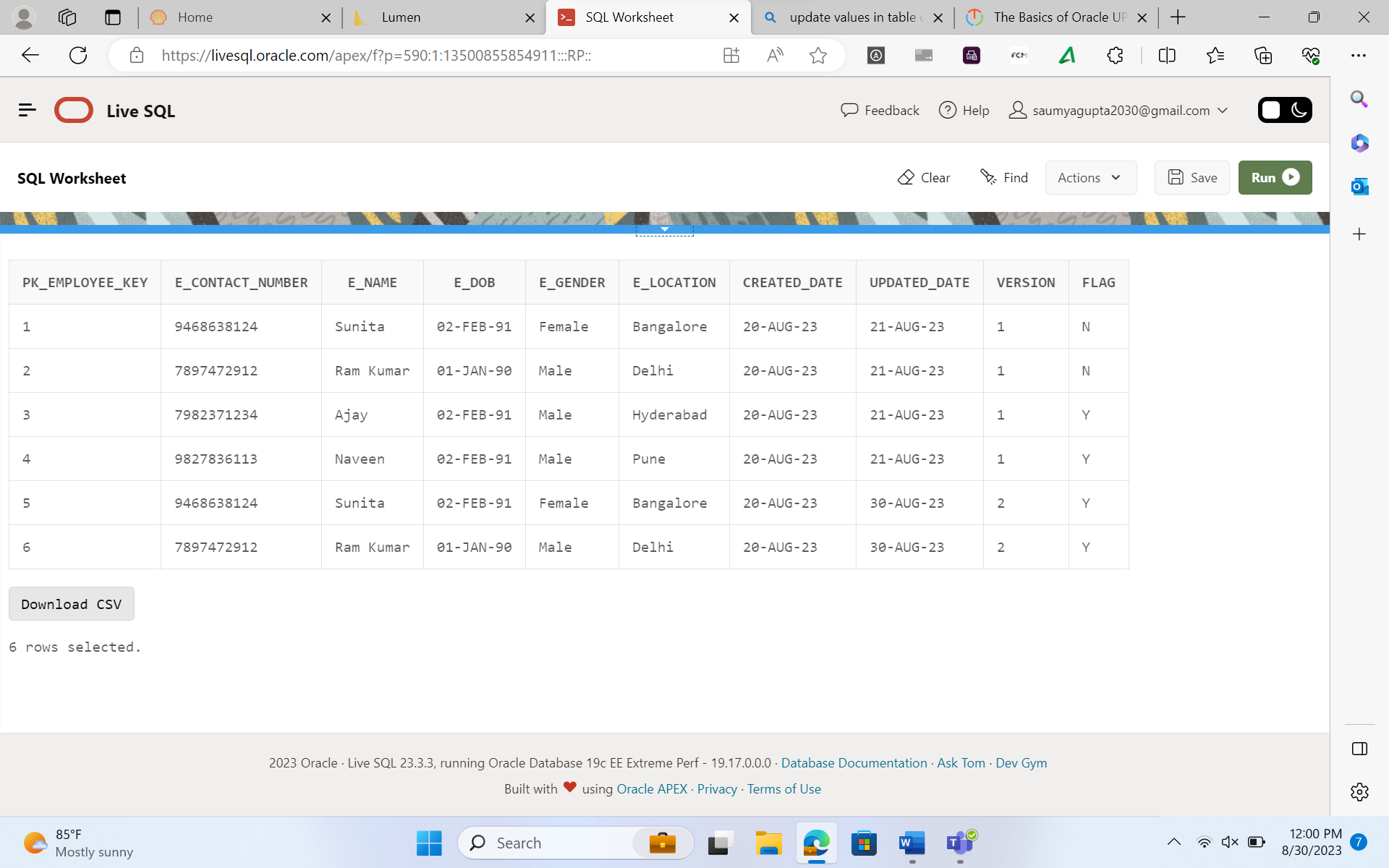
insert into D\_emp\_SCD2 values(2, 7897472912, 'Ram Kumar', '01-jan-1990','Male', 'Delhi','20-aug-2023', '21-aug-2023', 1, 'N');

insert into D\_emp\_SCD2 values(3, 7982371234, 'Ajay', '02-feb-1991', 'Male', 'Hyderabad','20-aug-2023', '21-aug-2023', 1, 'Y');

insert into D\_emp\_SCD2 values(4, 9827836113, 'Naveen', '02-feb-1991','Male', 'Pune','20-aug-2023', '21-aug-2023', 1, 'Y');

insert into D\_emp\_SCD2 values(5, 9468638124, 'Sunita', '02-feb-1991','Female', 'Bangalore','20-aug-2023', '30-aug-2023', 2, 'Y');

insert into D\_emp\_SCD2 values(6, 7897472912, 'Ram Kumar', '01-jan-1990','Male', 'Delhi','20-aug-2023', '30-aug-2023', 2, 'Y');



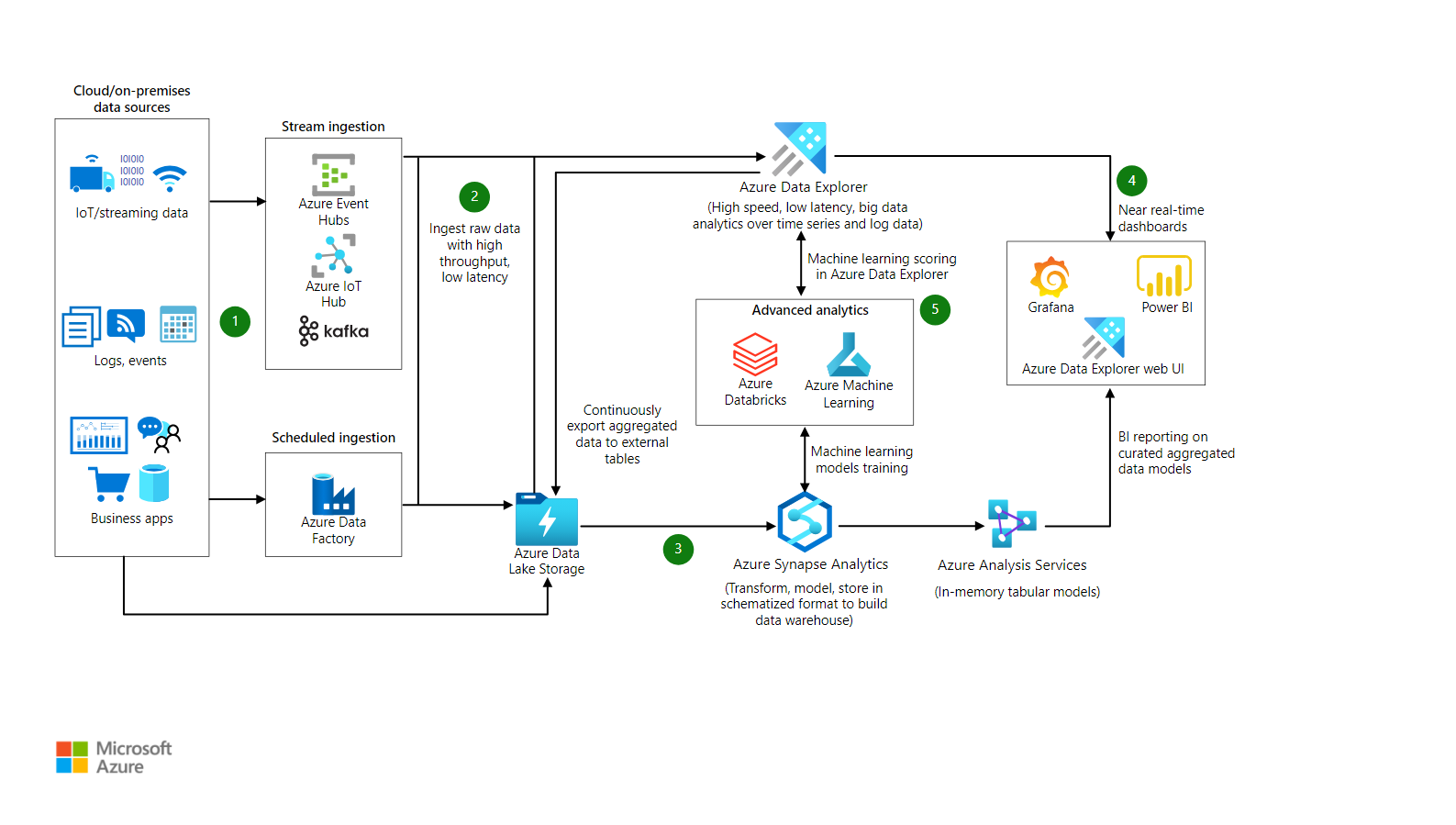
**Denormalisation:**

* It is a database design technique which involves intentionally introducing redundancy into a relational database schema to improve query performance.
* While normalization is the process of organizing the data to minimize redundancy and data anomalies, denormalization is used when the priority is optimizing the read performance for specific types of queries.

**Overview of BigData**

* BigData refers to the vast and complex volume of the data that exceeds the processing capacity of traditional database systems and requires specialized tools and techniques to store process and analyse affectively.
* It encompasses large datasets that are difficult to manage, process and analyse using traditional data processing methods.
* The 4 Vs of Big Data
  + Volume
  + Velocity
  + Variety
  + Veracity
* Challenges of Big Data
  + Storage & Management
  + Processing Power
  + Data Integration
  + Data Quality
  + Privacy and security
  + Analysis and Insights

**Big Data Architecture**

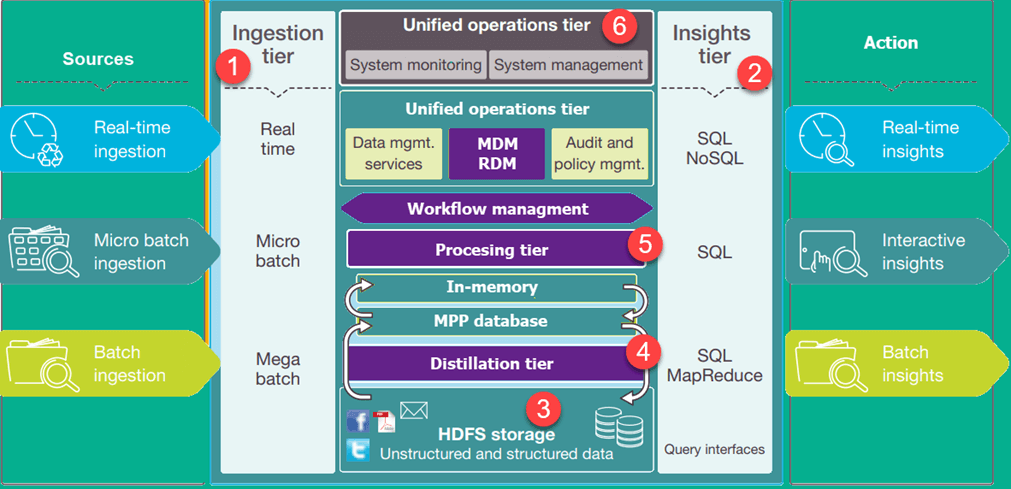


**Importance of Big Data**

* **Structured Data:** Traditional relational databases have been managing structured data for decades. The analysis is relatively straight forward making it useful for reporting and decision making.
* **Semi Structured Data:**
* **Unstructured Data:**

**Tools used in BigData**

* **Hadoop – HDFS** 
  + HDFS clusters contains single namenode and multiple DataNodes.
  + **File system namespace:** A user can store and access files in this directory.
  + HDFS is designed to reliably store v large files across machines in large clusters. It stores each files as sequence of blocks. The blocks of the files are replicated for fault tolerance.
* **Apache Spark**
  + Provides in memory data processing
  + Helps in real time streaming and ML.
* **NoSQL Databases:**
* **Apache Kafka:**
  + Distributed streaming platform used for realtime data pipelines and streaming applications.
  + Efficiently handles high volumes of streaming data and enables real time analytics.
* **ElasticSearch:** distributed seach and analytical engine that can handle and search large volumes of data
* **Kibana:** Data visualization tool
* **Data Lakes:**
* **Data Lake Architecture:**



* We can build data lake solutions using the following services offered by azure:
  + Azure HDInsight
  + Azure data lake store
  + Azure Data Lake Analytics